

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application.

Listing of Claims:

1. (Currently amended) A method for determining whether an input message associated with an input location point belongs to a display window associated with an application, wherein the window has been transformed on the display by a transformation matrix, the method comprising:

determining a bounding rectangle bounding the window transformed by the transformation matrix;

determining whether the input device point is within the bounding rectangle; [[and]]

performing an inverse transformation on the input device location point if the input device point is within the bounding rectangle, wherein performing the inverse transformation results in a new frame region; and[[.]]

performing a hit test on the new frame region.

2. (Original) The method of claim 1, wherein the performing operation further comprises:

determining whether the input device point is within a new frame region within the bounding rectangle; and

performing the inverse transformation on the input device location point only if the input device point is within the new frame region within the bounding rectangle.

3. (Original) The method of claim 2, wherein the new frame region is the bounding rectangle and is defined by a pair of triangles.

4. (Original) The method of claim 3, wherein the determining operation further comprises:

using the transform matrix to map the two triangles into transformed space; and

determining whether the input device point is within one of the two triangles.

5. (Original) The method of claim 1, wherein the input location point is a mouse input point.

6. (Original) The method of claim 1, wherein the inverse transformation converts the input device location point to window logical coordinate space for transmission to the application.

7. (Currently amended) A method for redirecting an input device message from an input device to an application via a transformed window on a display, wherein the transformed window is an application window that has been transformed on the display by a transformation matrix, the method comprising:

receiving the input device message from the input device at an input device location point on the display;

determining whether the location point is within the transformed window;

if the location point is within the transformed window, performing an inverse transform on the location point into its logical coordinate system coordinates; [[and]]

performing a hit test on a new frame region in the logical coordinate system; and

communicating the device message to the application.

8. (Original) The method of claim 7 wherein the determining operation comprises:

determining whether the input device point is within a bounding rectangle bounding the transformed window; and

performing the inverse transformation on the input device location point if the input device point is within the bounding rectangle.

9. (Original) The method of claim 8 wherein the determining operation further comprises:

determining whether the input device point is within a new frame region within the bounding rectangle; and

performing the inverse transformation on the input device location point only if the input device point is within the new frame region within the bounding rectangle.

10. (Original) The method of claim 9, wherein the new frame region is the bounding rectangle and the bounding rectangle is defined by two adjoining triangles.

11. (Original) The method of claim 10, wherein the determining operation further comprises:

using the transformation matrix to map the two triangles into transformed space; and
determining whether the input device point is within one of the two triangles.

12. (Currently amended) A system for maintaining an application window on a display that has been transformed by a transformation matrix into a transformed window transparent to an application that originated the application window, the system comprising:

a desktop window manager operable to transform a window transmitted by the application for display to a transformed window on the display;

a redirection transform application interface coupled to the manager, wherein the redirection transform application performs an inverse transformation that results in a new frame region and performs hit detection on the new frame region; and

an external input coupled to the desktop window manager, wherein messages to and from the application pass through the transform interface to and from the manager to maintain transparency between the transformed display window and the application.

13. (Original) The system of claim 12 wherein the window manager includes an input hit testing routine to determine whether an external input message to the manager is associated with a transformed window and perform an inverse transformation on the input if the input is associated with the transformed window before sending the input message to the application.

14. (Original) The system of claim 13, wherein the external input is a mouse input.

15. (Original) The system of claim 13, wherein the external input is an input message from another application.

16. (Original) The system of claim 13, wherein the input hit test routine comprises a speed hit test routine and a full hit test routine in order to timely process the external input message.

17. (Currently amended) A [[machine-readable]] computer storage medium encoding a computer program of instructions for performing a method for redirecting an input device message from an input device to an application via a transformed window on a display, wherein the transformed window is an application window that has been transformed on the display by a transformation matrix, by a computer system, said ~~computer program~~ method comprising:

determining a bounding rectangle bounding the window transformed by the transformation matrix;

determining whether the input device point is within the bounding rectangle; and

performing an inverse transformation on the input device location point if the input device point is within the bounding rectangle, wherein performing the inverse transformation results in a new frame region; and [[.]]

performing a hit test on the new frame region.

18. (Currently amended) The [[machine-readable]] computer storage medium of claim 17, wherein the performing operation further comprises:

determining whether the input device point is within a new frame region within the bounding rectangle; and

performing the inverse transformation on the input device location point only if the input device point is within the new frame region within the bounding rectangle.

19. (Currently amended) The [[machine-readable]] computer storage medium of claim 17, wherein the determining operation further comprises:

using the transform matrix to map two triangles defining the bounding rectangle into transformed space; and

determining whether the input device point is within one of the two triangles.

20. (Currently amended) The [[machine-readable]] computer storage medium of claim 17, the method further comprising:

communicating the device message to the application.